UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/824,816	04/14/2004	Alexander Joffe	M-5648-2D US	6681

32605 7590 12/08/2008

MACPHERSON KWOK CHEN & HEID LLP
2033 GATEWAY PLACE
SUITE 400
SAN JOSE, CA 95110

EXAMINER	
SPIELER, WILLIAM	
ART UNIT	**PAPER NUMBER**
2169	

MAIL DATE	DELIVERY MODE
12/08/2008	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	10/824,816	JOFFE ET AL.	
	Examiner	Art Unit	
	WILLIAM SPIELER	2169	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *30 September 2008*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *11-13 and 30-52* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *11-13 and 30-52* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 11-13 and 30-52 are pending.

2. All pending claims are rejected.

Claim Rejections - 35 USC § 102

3. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

4. Claims 11, 13, 43, 45, 47-52 are rejected under 35 U.S.C. 102(e) as being anticipated by Hines, U.S. PGPub 2003/0121027 (hereinafter "Hines").

As per Claim 11, Hines teaches:

a circuit for allocating each resource to the tasks in a continuous operation so that in said operation, after any one of the tasks has finished accessing any one of the resources in processing a data unit, said one of the tasks does not get access to the same resource until after every other one of the tasks has finished accessing the resource (Hines, ¶ 0128, "[In a] round-robin resource allocation protocol implementing a token passing scheme . . . a logical token 324 symbolizes the right to access resource 311, i.e., when a software element holds token 324, it has the *right* to

access resource 311. When one of software elements 312, 314, 316, 318, 320, or 322 *finishes with resource* 311, it passes token 324, and with token 324 the access right, to a successor." (emphasis added); Hines, Fig. 3B), where it is clear that the round-robin scheduling method of Hines does not pass the token until the task holding the token has, in fact, completed access to the resource by "finish[ing] with [the] resource";

wherein accessing any one of said resources by any one of said tasks comprises:

(1) the task attempting to access the resource, wherein attempting to access the resource comprises generating a signal indicating that the task is attempting to access the resource;

(2) in response to the operation (1), the circuit allowing the task to access the resource if the resource is available to the task, the circuit not allowing the task to access the resource until the resource becomes available to the task;

(3) the task accessing the resource when the circuit allows the task to access the resource (Hines, ¶ 0128, "[In a] round-robin resource allocation protocol implementing a token passing scheme . . . a logical token 324 symbolizes the right to access resource 311, i.e., when a software element holds token 324, it has the *right* to access resource 311"), where it is clear that without the token, the task will not be permitted to access the resource, and that the task will be permitted to access the resource once the task gains possession of the token;

wherein for any task T1 of said tasks and any resource R1 of said resources, if the task T1 attempts to access the resource R1 after the task T1 has already finished accessing the resource R1, and at least one other task T2 has not attempted to access the resource R1 after the task T1 has finished accessing the resource R1, then the circuit will never allow the task T1 to access the resource R1 in said operation until the task T2 attempts to access the resource R1 and accesses the resource R1 in said operation (Hines, ¶ 0128, "[In a] round-robin resource allocation protocol implementing a token passing scheme . . . a logical token 324 symbolizes the right to access resource 311, i.e., when a software element holds token 324, it has the *right* to access resource 311. When one of software elements 312, 314, 316, 318, 320, or 322 *finishes with resource* 311, it passes token 324, and with token 324 the access right, to a successor." (emphasis added); Hines, Fig. 3B), where it is clear that as the round-robin scheduling method of Hines requires the token to be passed to all other tasks in the ring before a given task will receive the token again thereby permitting access to the resource and that a task does not pass the token until it has accessed the resource, the claimed functionality will result.

As per Claim 43, the rejection of Claim 11 is incorporated, and Hines teaches:

at least one of the resources is accessed multiple times by each of said tasks (Hines, ¶ 0128, "[In a] round-robin resource allocation protocol implementing a token passing scheme . . . a logical token 324 symbolizes the right to access resource

311, i.e., when a software element holds token 324, it has the right to access resource 311. When one of software elements 312, 314, 316, 318, 320, or 322 finishes with resource 311, it passes token 324, and with token 324 the access right, to a successor."; Hines, Fig. 3B), where it is clear from Figure 3B that a given task will access the resource multiple times as the token is passed around the cycle.

Claims 13 and 45 correspond to Claims 11 and 43 respectively and are rejected for the same reasons set forth in connection with their respective claims above.

As per Claim 47, Hines teaches:

a circuit for allocating each resource to the tasks in a continuous operation so that in said operation, after any one of the tasks has finished accessing any one of the resources in processing a data unit, said one of the tasks does not get access to the same resource until after every other one of the tasks has finished accessing the resource (Hines, ¶ 0128, "[In a] round-robin resource allocation protocol implementing a token passing scheme . . . a logical token 324 symbolizes the right to access resource 311, i.e., when a software element holds token 324, it has the *right* to access resource 311. When one of software elements 312, 314, 316, 318, 320, or 322 *finishes with resource* 311, it passes token 324, and with token 324 the access right, to a successor." (emphasis added); Hines, Fig. 3B), where it is clear that the round-robin scheduling method of Hines does not pass the token until the task holding the token has, in fact, completed access to the resource by "finish[ing] with [the] resource";

wherein accessing any one of said resources by any one of said tasks comprises:

(1) the task attempting to access the resource, wherein attempting to access the resource comprises generating a signal indicating that the task is attempting to access the resource;

(2) in response to the operation (1), the circuit allowing the task to access the resource if the resource is available to the task, the circuit not allowing the task to access the resource until the resource becomes available to the task;

(3) the task accessing the resource when the circuit allows the task to access the resource (Hines, ¶ 0128, "[In a] round-robin resource allocation protocol implementing a token passing scheme . . . a logical token 324 symbolizes the right to access resource 311, i.e., when a software element holds token 324, it has the *right* to access resource 311"), where it is clear that without the token, the task will not be permitted to access the resource, and that the task will be permitted to access the resource once the task gains possession of the token;

wherein for any task T1 of said tasks and any resource R1 of said resources, if the task T1 attempts to access the resource R1 after the task T1 has already finished accessing the resource R1, and at least one other task T2 has not attempted to access the resource R1 after the task T1 has finished accessing the resource R1, then the circuit will never allow the task T1 to access the

resource R1 in said operation until the task T2 attempts to access the resource R1 and accesses the resource R1 in said operation (Hines, ¶ 0128, "[In a] round-robin resource allocation protocol implementing a token passing scheme . . . a logical token 324 symbolizes the right to access resource 311, i.e., when a software element holds token 324, it has the *right* to access resource 311. When one of software elements 312, 314, 316, 318, 320, or 322 *finishes with resource* 311, it passes token 324, and with token 324 the access right, to a successor." (emphasis added); Hines, Fig. 3B), where it is clear that as the round-robin scheduling method of Hines requires the token to be passed to all other tasks in the ring before a given task will receive the token again thereby permitting access to the resource and that a task does not pass the token until it has accessed the resource, the claimed functionality will result.

As per Claim 48, the rejection of Claim 47 is incorporated, and Hines teaches:

at least one of the resources is accessed multiple times by each of said tasks (Hines, ¶ 0128, "[In a] round-robin resource allocation protocol implementing a token passing scheme . . . a logical token 324 symbolizes the right to access resource 311, i.e., when a software element holds token 324, it has the right to access resource 311. When one of software elements 312, 314, 316, 318, 320, or 322 finishes with resource 311, it passes token 324, and with token 324 the access right, to a successor."; Hines, Fig. 3B), where it is clear from Figure 3B that a given task will access the resource multiple times as the token is passed around the cycle.

As per Claim 49, the rejection of Claim 47 is incorporated, and Hines further teaches:

in processing each of said data units, the corresponding one of the tasks accesses the resources one after another in a predefined sequence (Hines, ¶ 0128, "[In a] round-robin resource allocation protocol implementing a token passing scheme . . . a logical token 324 symbolizes the right to access resource 311, i.e., when a software element holds token 324, it has the right to access resource 311. When one of software elements 312, 314, 316, 318, 320, or 322 finishes with resource 311, it passes token 324, and with token 324 the access right, to a successor."; Hines, Fig. 3B), where it is clear that there is a predefined sequence in the succession of the token.

Claims 50-52 correspond to Claims 47-49 respectively and are rejected for the same reasons set forth in connection with their respective claims above.

Claim Rejections - 35 USC § 103

5. The text of those sections of Title 35, U.S. Code not included in this action can be found in a prior Office action.

6. Claims 12 and 35 are rejected under 35 U.S.C. § 103(a) as being unpatentable over Hines, U.S. PGPub 2003/0121027 (hereinafter "Hines") in view of Bahr et al., U.S. Patent No. 5,167,022 (hereinafter "Bahr").

As per claim 12, the rejection of claim 11 is hereby incorporated, and further Hines does not teach that "for at least one resource, each task starts accessing the resource by locking the resource to make it unavailable to any other task, and the task finishes accessing the resource by unlocking the resource." The analogous art of Bahr teaches the use of a mutual exclusion lock - or mutex - to accomplish this task.

Therefore it would have been obvious to one of the ordinary skill in the art at the time of invention was made to incorporate the teaching of Bahr into the method of Hines to use a mutual exclusion lock because one of the ordinary skill in the art would want to lock the resources in this manner by way of a mutex, which would serve to ensure that the system remains stable as resources are being accessed in a predictable manner in a multi-tasking computer system.

Claim 35 is the method claim corresponding to the system claim 12, and is rejected under the same reason set forth in connection to the rejection of claim 12 above.

7. Claims 30, 36 and 44 are rejected under 35 U.S.C. § 103(a) as being unpatentable over Hines, U.S. PGPub 2003/0121027 (hereinafter "Hines") in view of Nakade et al., U.S. Patent No. 4,847,751 (hereinafter "Nakade").

As per claim 30, the rejection of claim 11 is hereby incorporated, and further Hines does not teach that "each data unit is processed by a single one of the tasks which accesses at least two of said resources to process at least one of the data units." The analogous art of Nakade, however, does. (Abstract).

Therefore it would have been obvious to one of the ordinary skill in the art at the time of invention was made to incorporate the teaching of Nakade into the method of Hines because one of the ordinary skill in the art would access at least two resources (namely the processor and memory) to process a data unit.

Claim 36 is the method claim corresponding to the system claim 30, and is rejected under the same reason set forth in connection to the rejection of claim 30 above.

As per Claim 44, the rejection of Claim 30 is incorporated, and Hines further teaches:

in processing each of said data units, the corresponding one of the tasks accesses the resources one after another in a predefined sequence (Hines, ¶ 0128, "[In a] round-robin resource allocation protocol implementing a token passing scheme . . . a logical token 324 symbolizes the right to access resource 311, i.e., when a software element holds token 324, it has the right to access resource 311. When one of software elements 312, 314, 316, 318, 320, or 322 finishes with resource 311, it

passes token 324, and with token 324 the access right, to a successor."; Hines, Fig. 3B), where it is clear that there is a predefined sequence in the succession of the token.

8. Claims 31-33, 37-39, 41-42 and 46 are rejected under 35 U.S.C. 103(a) as being unpatentable over Hines, U.S. PGPub 2003/0121027 (hereinafter "Hines") in view of Nakade et al., U.S. Patent No. 4,847,751 (hereinafter "Nakade") and Sheth et al., U.S. Patent No. 5,386,517 (hereinafter "Sheth").

As per claim 31, the rejection of claim 30 is hereby incorporated, and further neither Hines nor Nakade teach that at least two of the resources accessed are storage area. The analogous art of Sheth teaches that a data processing task may access both a storage buffer and a command queue (Col. 24, lines 31-46).

Therefore it would have been obvious to one of the ordinary skill in the art at the time of invention was made to incorporate the teaching of Sheth into the method of Nakade and Hines because one of the ordinary skill in the art would use the storage buffers and command queues store data unit processing information of multiple data units, and that these storage areas could be accessed by a task, and that having storage areas would be of obvious utility in processing data units.

As per claim 32, the rejection of claim 31 is hereby incorporated, and the analogous art of Sheth teaches that a data processing task may access both a storage buffer and a command queue (Col. 24, lines 31-46). Sheth further teaches reading the

request from the request FIFO (Col. 24, lines 34-35) and writing one or more commands to the command FIFO (Col. 24, lines 43-45).

Therefore it would have been obvious to one of the ordinary skill in the art at the time of invention was made to incorporate the teaching of Sheth into the method of Hines because one of the ordinary skill in the art would recognize that FIFOs are an easier form of data storage to implement a a simple queue/dequeue operation as opposed to more complicated pointers and references, and that these are of particular use in serially-transmitted data such as requests and associated commands

As per claim 33, the rejection of claim 32 is hereby incorporated, and further the analogous art of Sheth teaches that requests can be made to an address. (Col. 28, line 38).

Therefore it would have been obvious to one of the ordinary skill in the art at the time of invention was made to incorporate the teaching of Sheth into the method of Hines because one of the ordinary skill in the art that the address requested may be contained in the request as claimed, and that this would be a manner to encapsulate request information into a single object.

As per claim 41, the rejection of claim 31 is incorporated and further Sheth teaches:

the data processing information is associated with an order of data units, the order being the same for each said storage area (Col. 24, lines 31-45). Sheth

teaches reading a message off a message handler FIFO (Col. 24, lines 31-33) then writing a message command to a system bus command FIFO (Col. 24, lines 43-45). Given the nature of a FIFO, the order of the data units on the respective FIFOs will inherently be the same: if a given data unit M_1 is after a second data unit M_2 on the message handler FIFO, these data units' respective data units on the command FIFO will be in the same order.

Claims 37-39 and 42 are the method claims corresponding to the system claim 31-33 and 41 respectively, and each is rejected under the same reason set forth in connection to the rejection of their corresponding claim above.

As per Claim 46, the rejection of Claim 37 is incorporated, and Hines further teaches:

in processing each of said data units, the corresponding one of the tasks accesses the resources one after another in a predefined sequence (Hines, ¶ 0128, "[In a] round-robin resource allocation protocol implementing a token passing scheme . . . a logical token 324 symbolizes the right to access resource 311, i.e., when a software element holds token 324, it has the right to access resource 311. When one of software elements 312, 314, 316, 318, 320, or 322 finishes with resource 311, it passes token 324, and with token 324 the access right, to a successor."; Hines, Fig. 3B), where it is clear that there is a predefined sequence in the succession of the token.

Response to Arguments

9. Applicant's arguments with respect to claims 11-13 and 30-52 have been considered but are moot in view of the new ground(s) of rejection.

Conclusion

10. Any inquiry concerning this communication or earlier communications from the examiner should be directed to WILLIAM SPIELER whose telephone number is (571) 270-3883. The examiner can normally be reached on Monday to Thursday, 11 AM - 1 PM Eastern.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, James Trujillo can be reached on (571) 272-3677. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/William Spieler/
Examiner, Art Unit 2169

/James Trujillo/
Supervisory Patent Examiner, Art Unit 2169